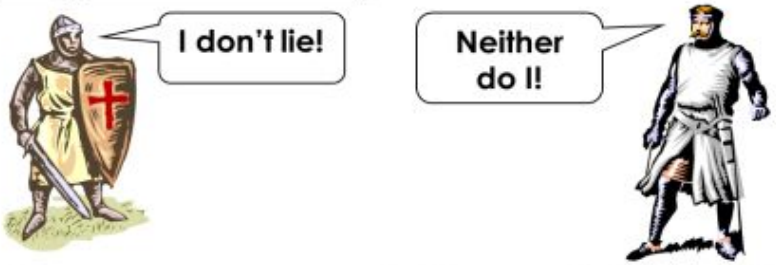
Nested if Statements and Logic

BOOLEAN LOGIC
if statements within another if statement

Logic Puzzle #1

Logic Warm Up



Suppose you are visiting an island with knights, who always tell the truth, and knaves, who always lie.

Question - Which statement is impossible for an islander to make?

1. I am a knight or a knave
2. I am a knight
3. I am a knave

Logic Puzzle #2

Third type of islander added: Spy can lie or tell the truth.

There is one spy, one knight, and one knave in this scenario
A says that C is a knave.
B says that A is a knight.
C says "I am the spy."

Which one is the spy, which one is the knight, which one is the knave? What can we rule out right away?

Logic Puzzle #2

Third type of islander added: Spy can lie or tell the truth.

There is one spy, one knight, and one knave in this scenario
A says that C is a knave.
B says that A is a knight.
C says "I am the spy."

Answer:
C is a knave.
A is telling the truth, so A is a knight.
B is a spy.

Boolean Logic

Did you know?
George Boole is the inventor of boolean logic. He was born in Lincoln, England and he was the son of a shoemaker in a low class family.

He was an English mathematician, philosopher, educator, and logician. He wrote the book called "The Laws of Thought" (contains boolean algebra)

George Boole,
(1815-1864)

Boolean Operators

There are two commonly used boolean operators in Python to **combine** conditions

- **and**
- **or**

Consider the following…

- To attain a letter grade of "B" in this class you must attain 70% - 80% exclusive.
- That is:

  If the mark is >= 70 AND the mark is < 80:

  You will attain a "B"

- In this example you will attain a "B" ONLY if BOTH conditions are met.
  - That is: mark is both >=70 and <80

‘and’ syntax

```
if (mark>=70) and (mark<80):
    print (“You have a B”)
```

Consider the following…

- To watch “Game of Thrones” you can either watch it on online OR you can watch it on HBO.
- That is:

  If you have a valid stream or have access to HBO:

  You can watch “Game of Thrones”
- In the previous example you can watch “Game of Thrones” if **at least** one of the conditions are met.
  - That is: method = = “online” OR method = = “HBO”

‘or’ syntax

```
if (method = = “online”) or (method = = “HBO”):
    print “You can watch Game of Thrones!”
    print “Yay!”
print “the end”
```

Operator	Description	Example
and	Returns true if both operands are true	(a == 3) and (b > 3) returns true
or	Returns true if at least one operand is true	(a < 2) or (b > 3) returns true
not	Reverses the logical state of a boolean expression.	not (a == 4) returns true

Recall:

- We use the if statement to choose between two alternatives
- What if we wanted more options (or alternatives) after the first alternatives?

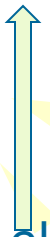
- Nested if statements involve placing an if statement within another if statement.

For Example…..

```
if (x > 0):
    if (y > 0):
        print (“This is printed if x>0 and then y>0”)
    else:
        print (“This is printed if x>0 and the y is not > 0”)
else:
    print (“This is printed if x is not > 0”)
    if (y > 0):
        print (“This is printed if x is not >0 and then y> 0”)
print (“The end”)
```

For Example…..

```
if (x > 0):
    if (y > 0):
        print ("This is printed if x>0 and then y>0")
    else:
        print ("This is printed if x>0 and the y is not > 0")
else:
    print ("This is printed if x is not > 0")
    if (y > 0):
        print ("This is printed if x is not >0 and then y> 0")
print ("The end")
```



- note that each else (or elif) statement is indented to be lined up with the if statement that it belongs to.
- Each nested if–elif-else construct is completed within their block

For example

Rewrite the following using a nested structure:

```
mark = float(input("Enter mark\n"))
message = " "
if mark >= 80:
        message="Honours"
elif mark >= 50:
        message="you pass"
else:
        message="you fail"
print (message)
```

```
mark = float(input("Enter mark"))
message = “error"
if mark >= 0:
        message ="you fail"
        if mark >= 50:
            message="you pass"
            if mark >= 80:
                message="honours“
                    if mark> 100:
                        message=“error”
print (message)
```